

SAZ Server/Service Update

19-Apr-2010

Keith Chadwick,
Neha Sharma,
Steve Timm

Why a new SAZ Server?

- Current SAZ server (V2_0_1b) has shown itself extremely vulnerable to user generated authorization “tsunamis”:
 - Very short duration jobs
 - User issues condor_rm on a large (>1000) glidein.
- This is fixed in the new SAZ Server (V2_7_0) using tomcat and a pools of execution and hibernate threads.
- We have found and fixed various other bugs in the current SAZ server and sazclient.
- We want to add support for the XACML protocol (used by Globus).
 - We will NOT transition to using XACML (yet).

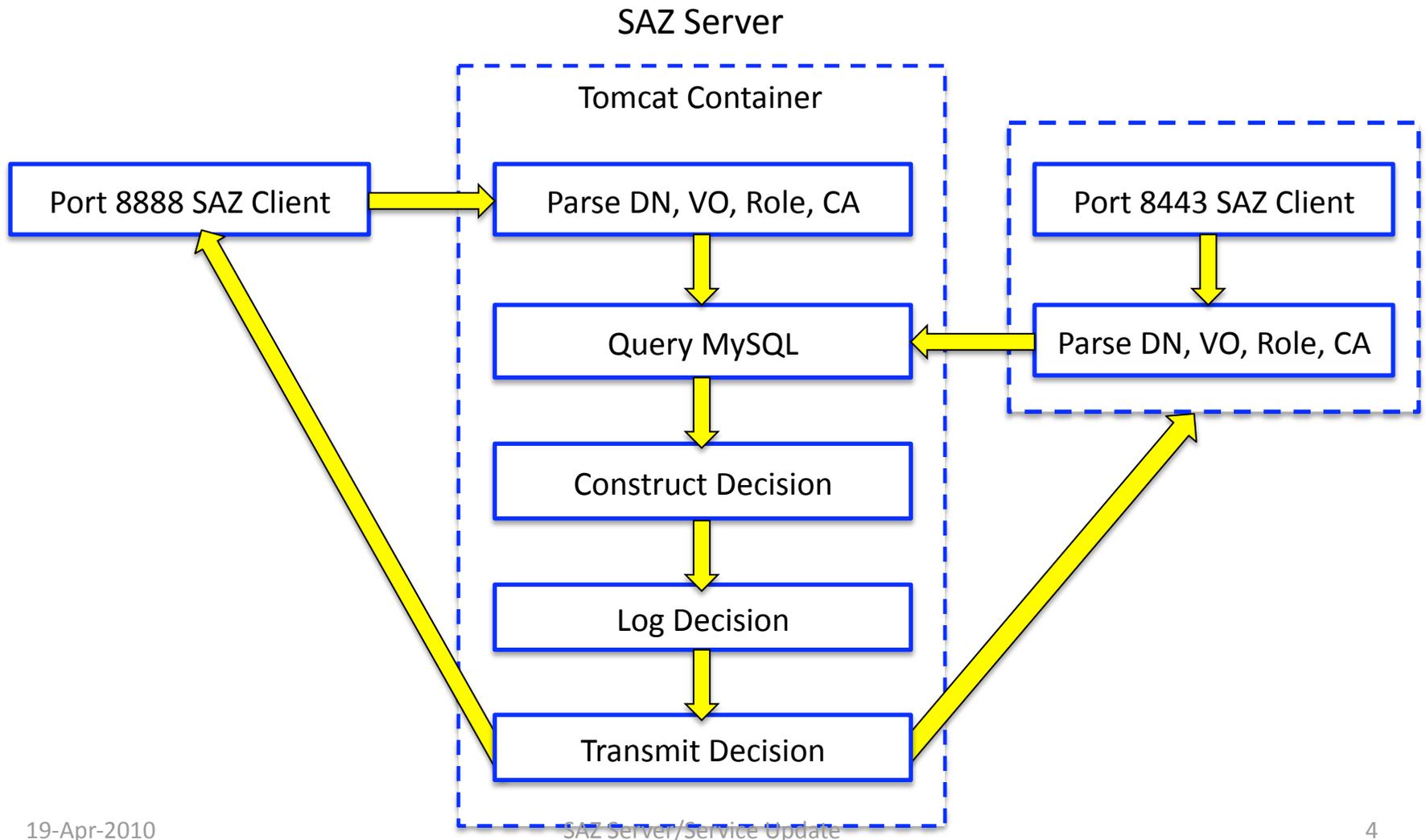
Old (“current”) SAZ Protocol – Port 8888

- Client sends the “entire” proxy to the SAZ server via port 8888.
- Server parses out DN, VO, Role, CA.
 - In SAZ V2.0.0b, the parsing logic does not work well, and frequently the SAZ server has to invoke a shell script voms-proxy-info to parse the proxy.
 - In the new SAZ V????, the parsing logic has been completely rewritten, and it no longer has to invoke the shell script voms-proxy-info to parse the proxy.
- Server performs MySQL queries.
- Server constructs the answer and sends it to the client.

New SAZ (XACML) Protocol – Port 8443

- Client parses out DN, VO, Role, CA and sends the information via XACML to the SAZ server via port 8443.
- Server performs MySQL queries.
- Server constructs the answer and sends it to the client.
- The new SAZ server supports both 8888 and 8443 protocols simultaneously.

Comparison of Old & New Protocol



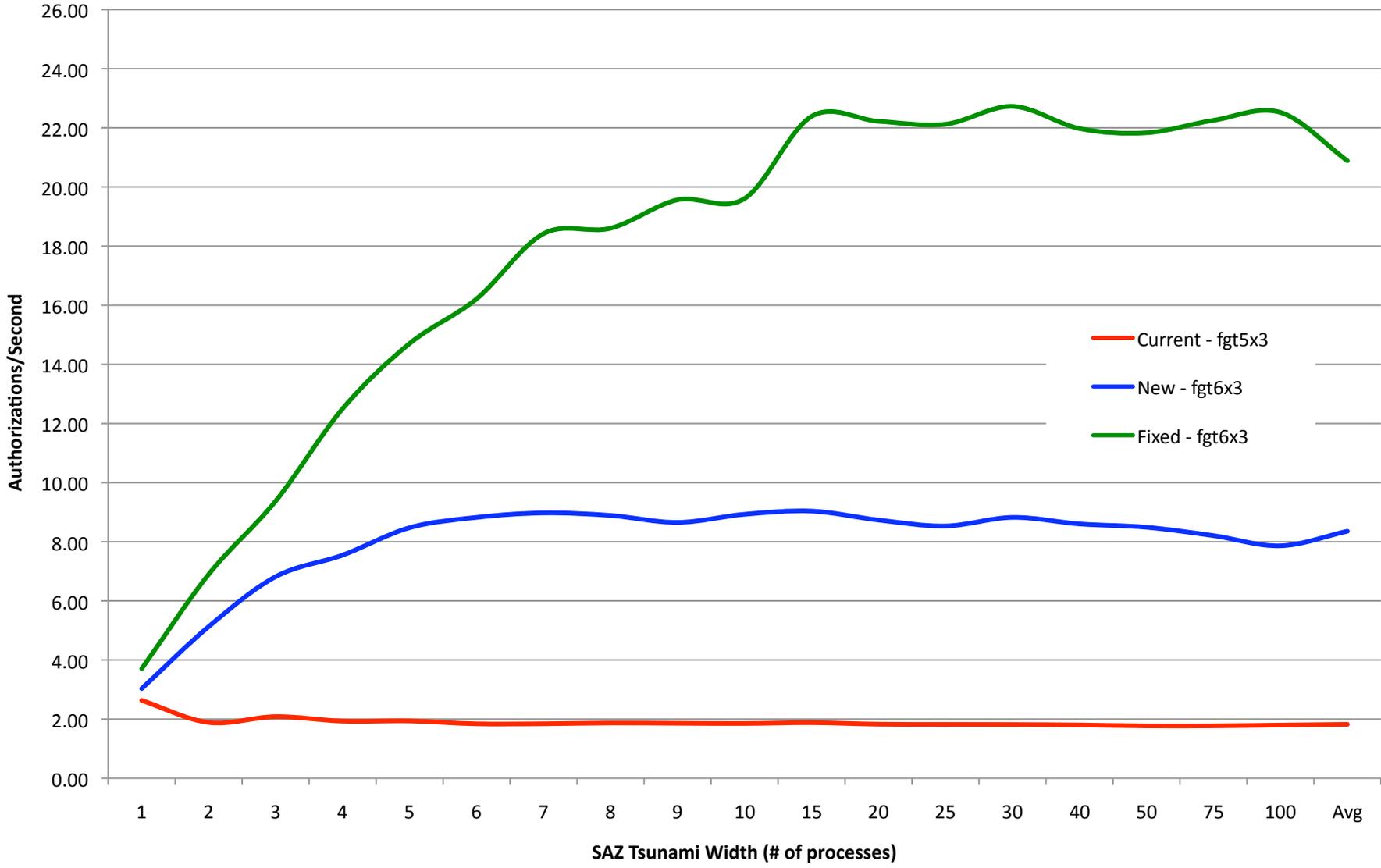
Some Definitions

- Width = # of processes doing SAZ calls/slot or system.
- Depth = # of SAZ calls.

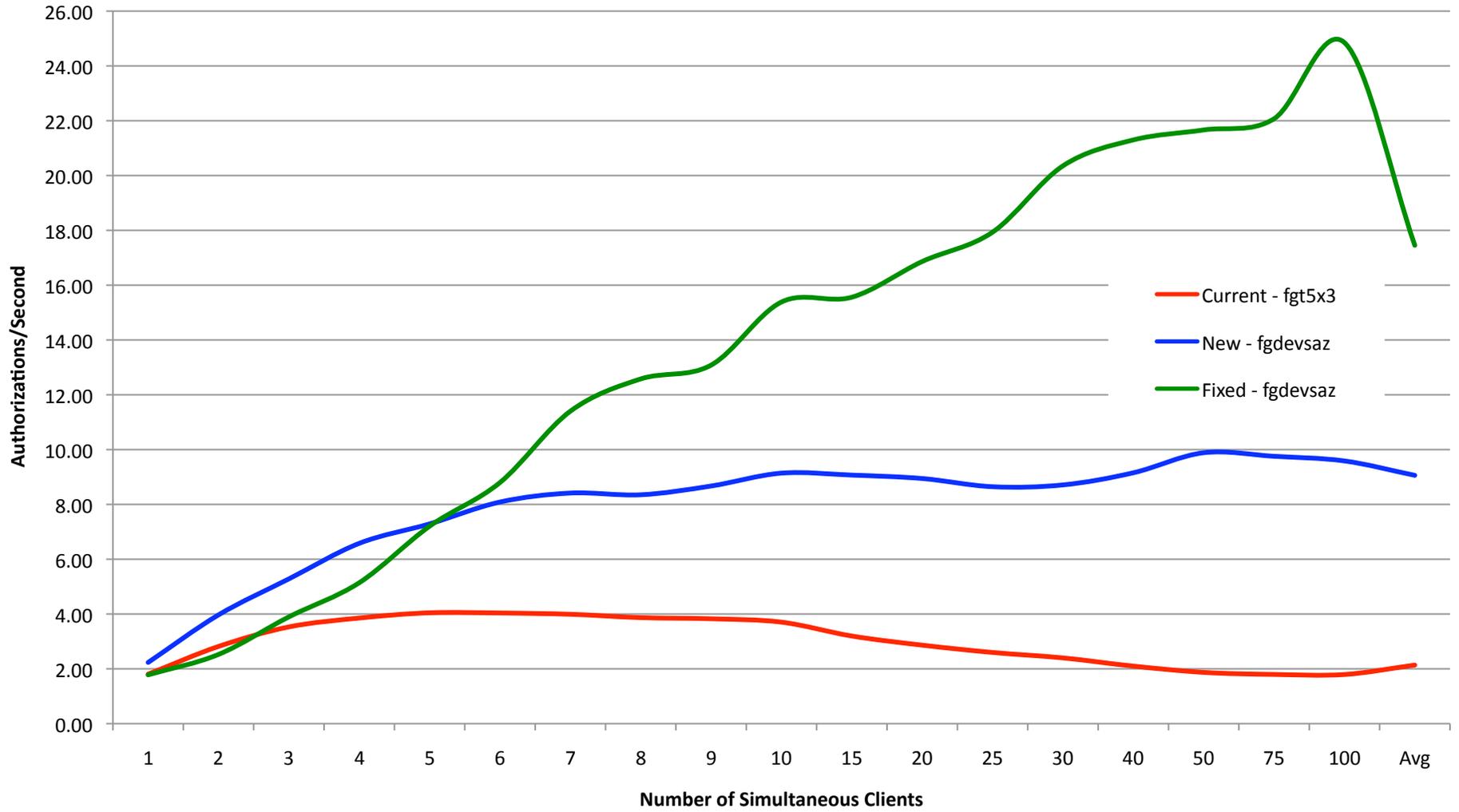
- Current = SAZ V2.0.0b
 - Currently deployed version of SAZ server.
- New = New SAZ Server
 - It handled small authorization tsunamis well
 - It was vulnerable to large (~1000) authorization tsunamis, (running out of file descriptors).
- Fixed = “Fixed” New SAZ Server
 - It has handled extra large (5000) authorization tsunamis without incident (ulimit 65535 to deal with the large number of open files).
 - It also has a greatly improved CRL access algorithm.

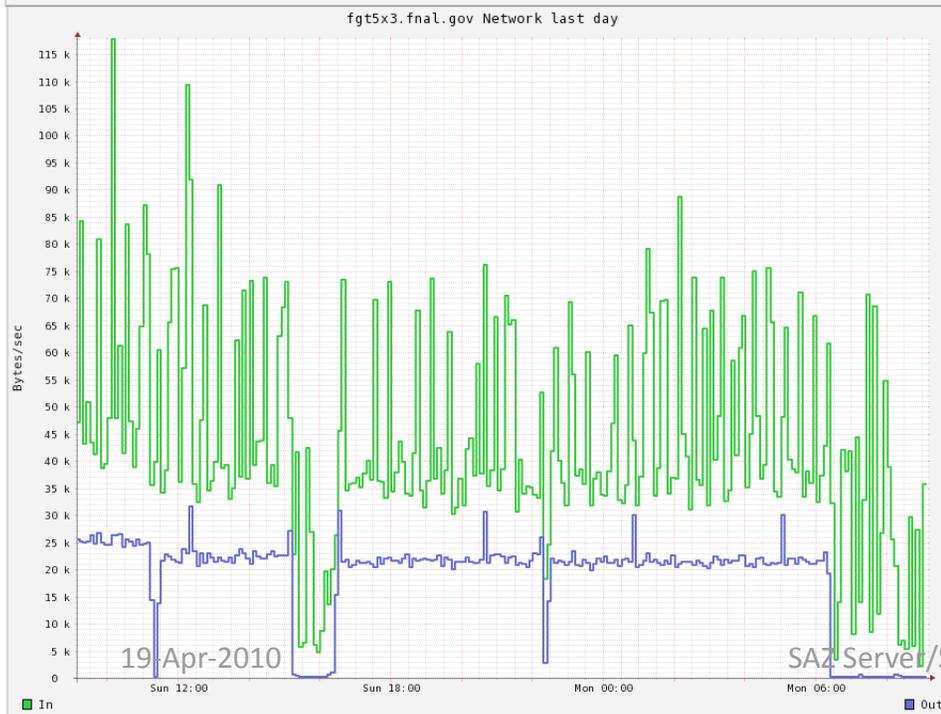
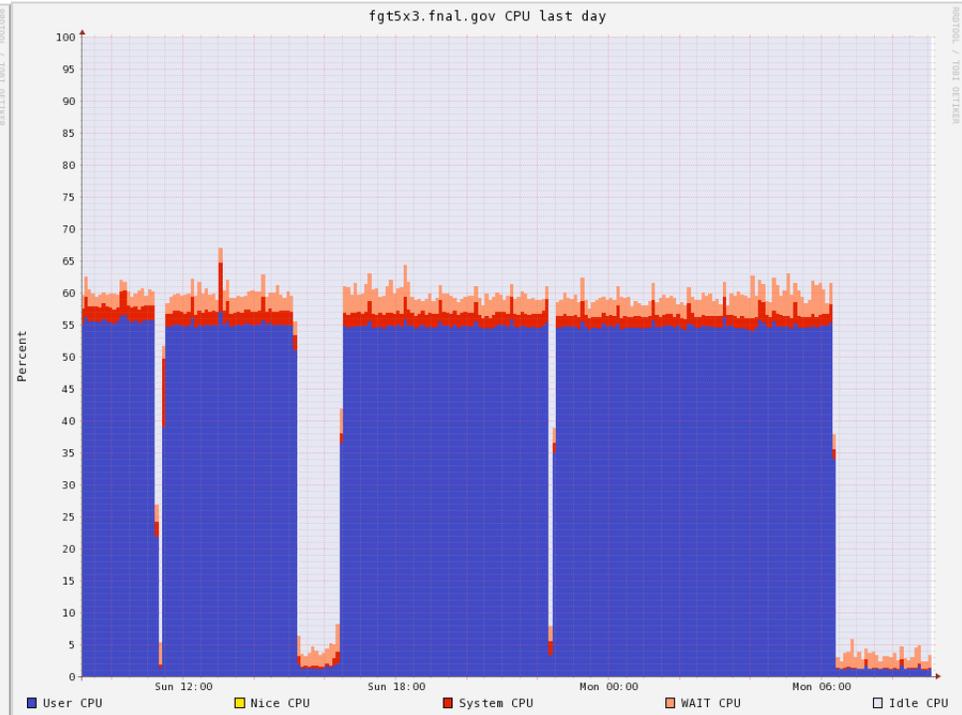
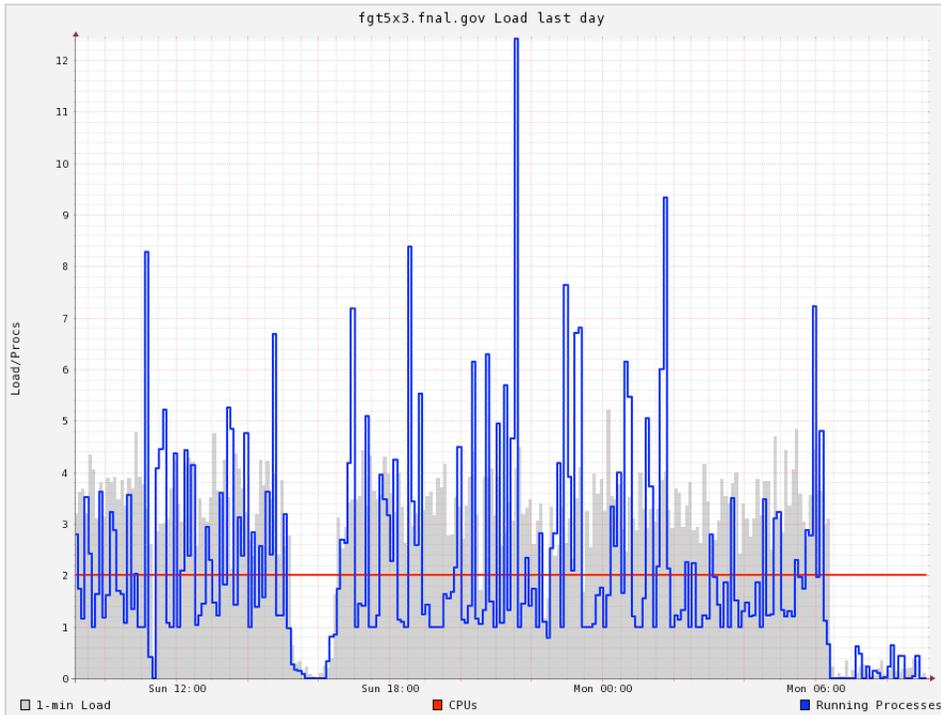
- All of the tests are run on/against SAZ servers on the fgtest systems:
 - fgtest[0-6] are FORMER production systems, 4+ years old, non-redundant.
 - Current production systems are at least 2x faster and redundant.

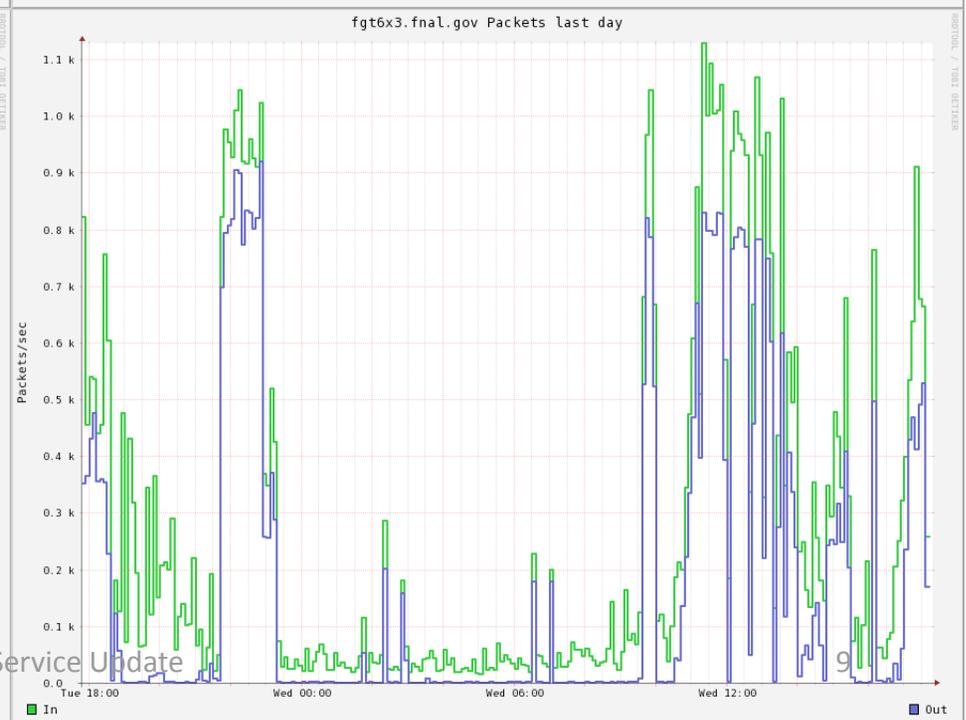
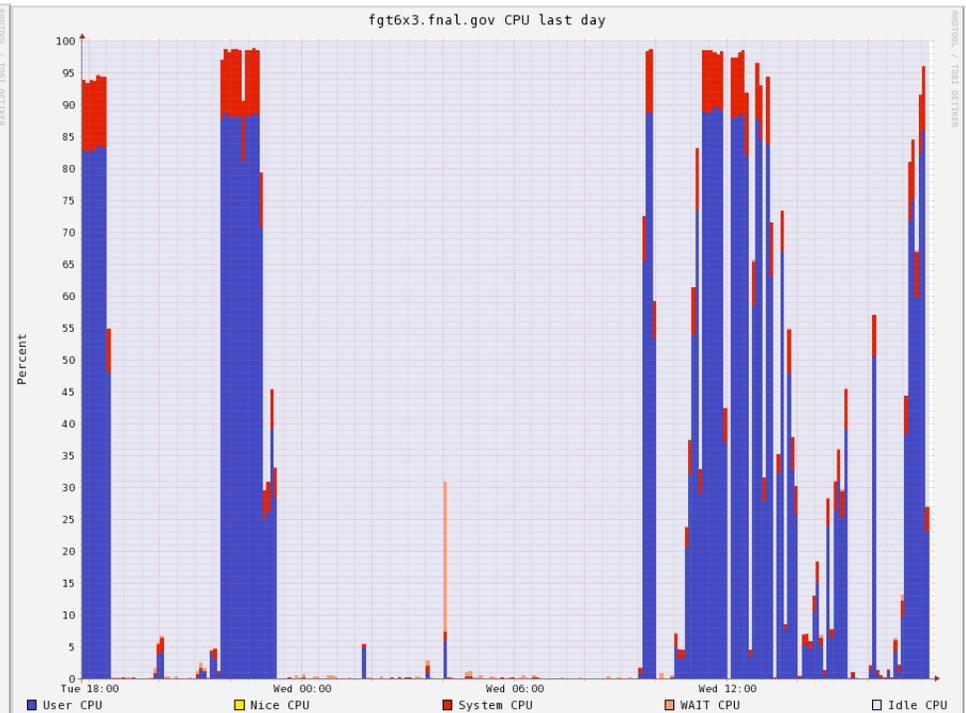
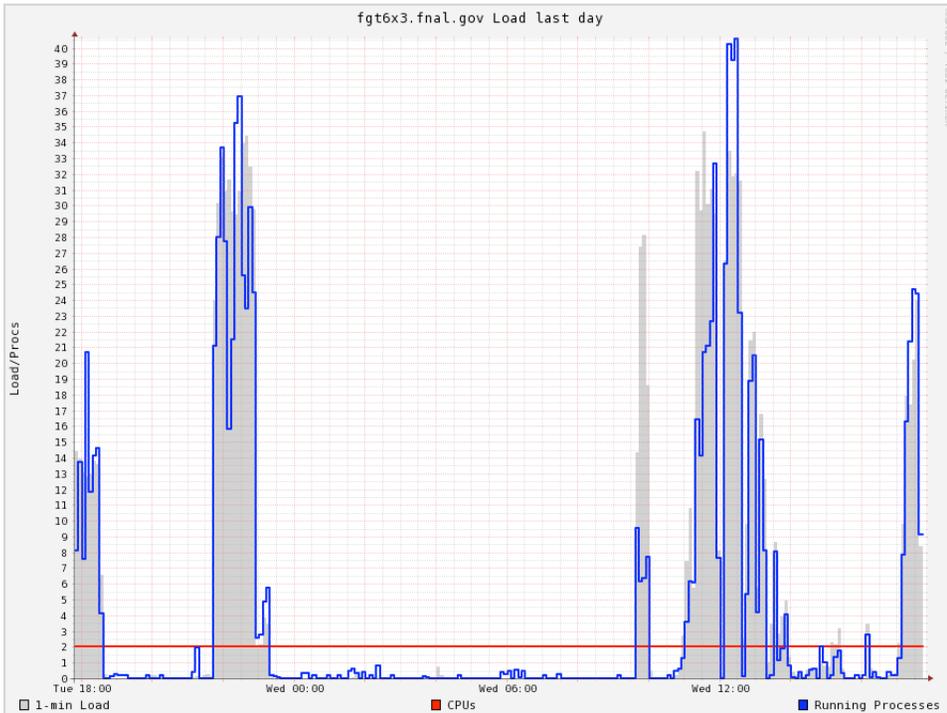
Single SAZ Server Performance - Single Client System - 100 Calls/Process



Single SAZ Server - 1 x 500 Calls/Client







19-Apr-2010 SAZ Server/Service Update 9

Tsunami Testing

- Using fgtest systems (4+ years old).
- Submit the first set of SAZ tests:
 - jobs=50, width=1, depth=1000
- Wait until all jobs are running.
- Trigger authorizations of the first test set.

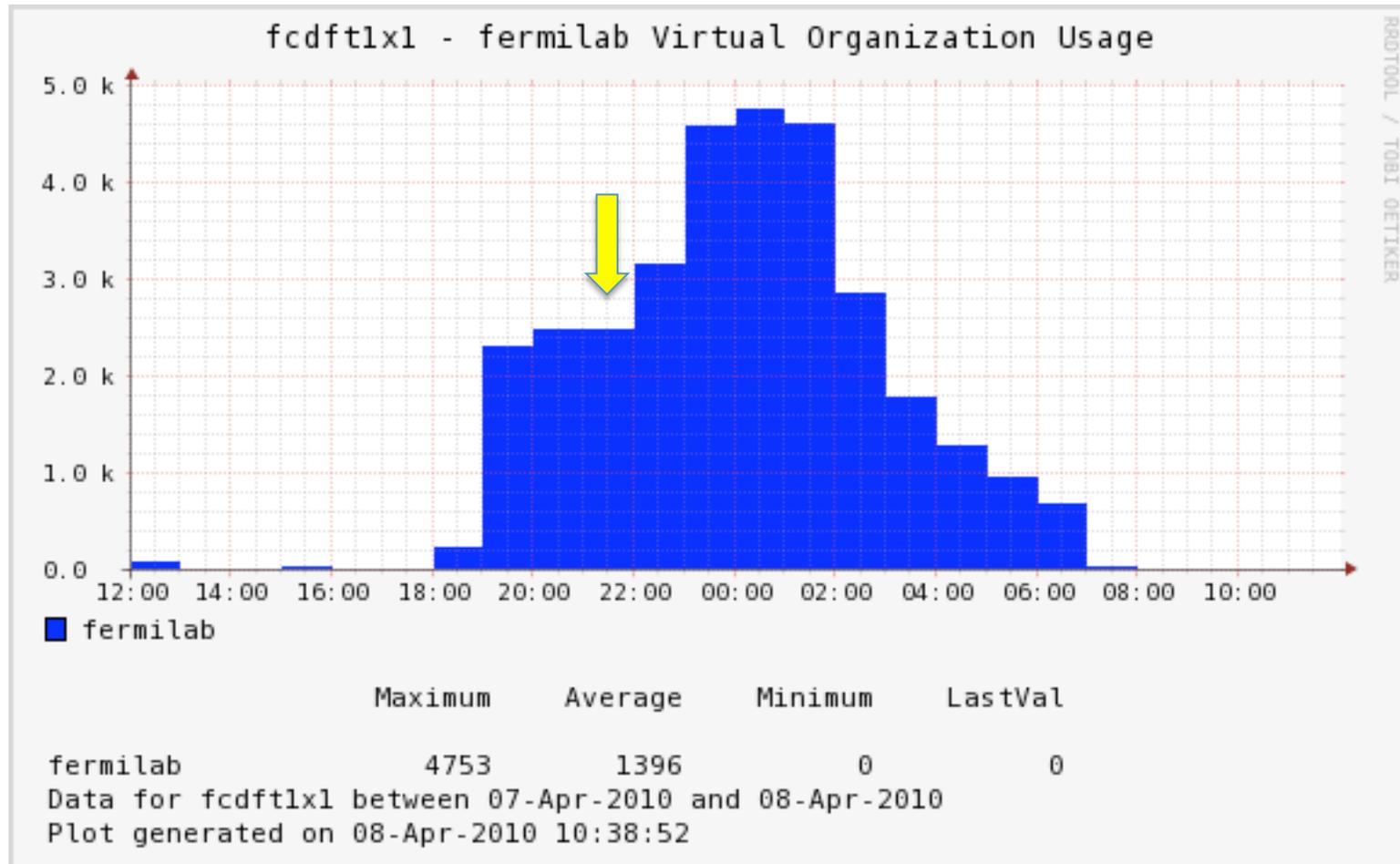
- Submit the second set of SAZ tests, either:
 - Jobs=1000, width=1, depth=50
 - Jobs=5000, width=1, depth=50
- Wait until all jobs are running.
- Trigger authorizations of the second test set.

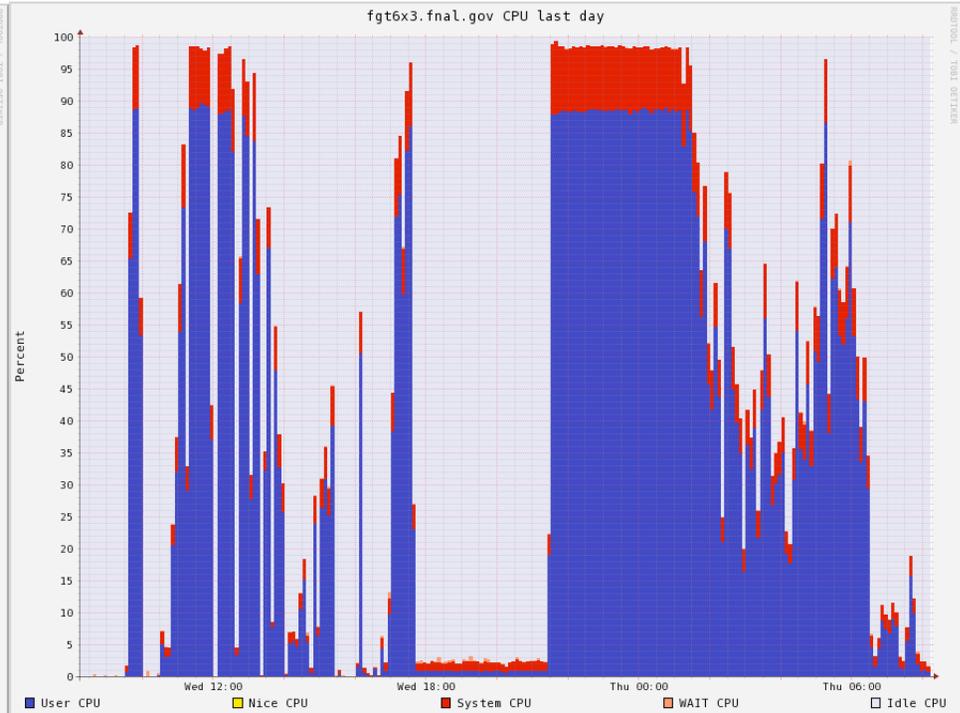
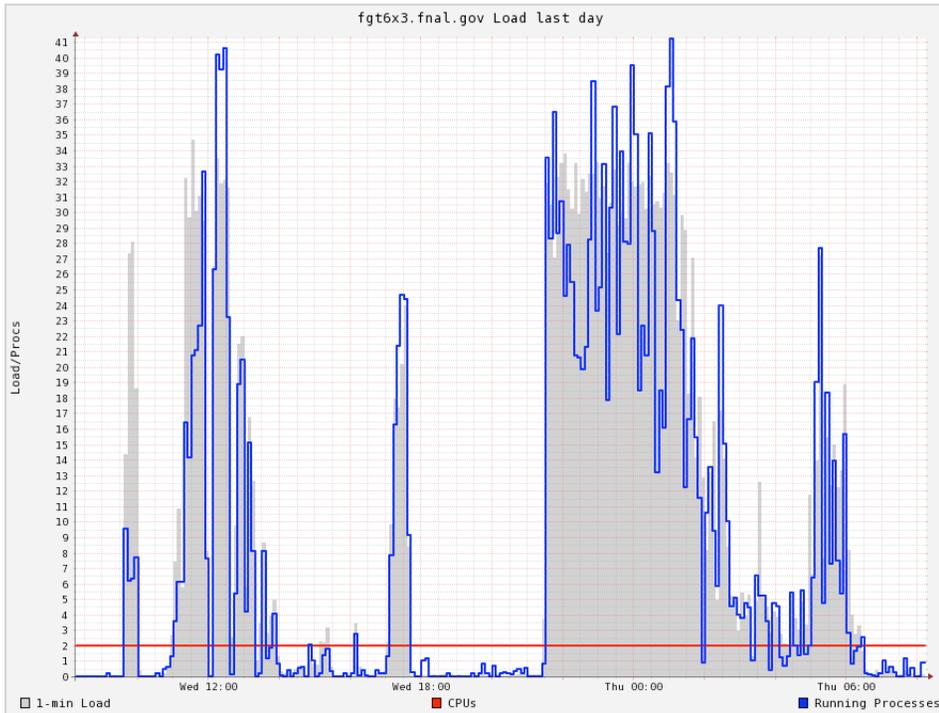
- Measure elapsed time for first and second sets

Results of the SAZ Tsunami Tests

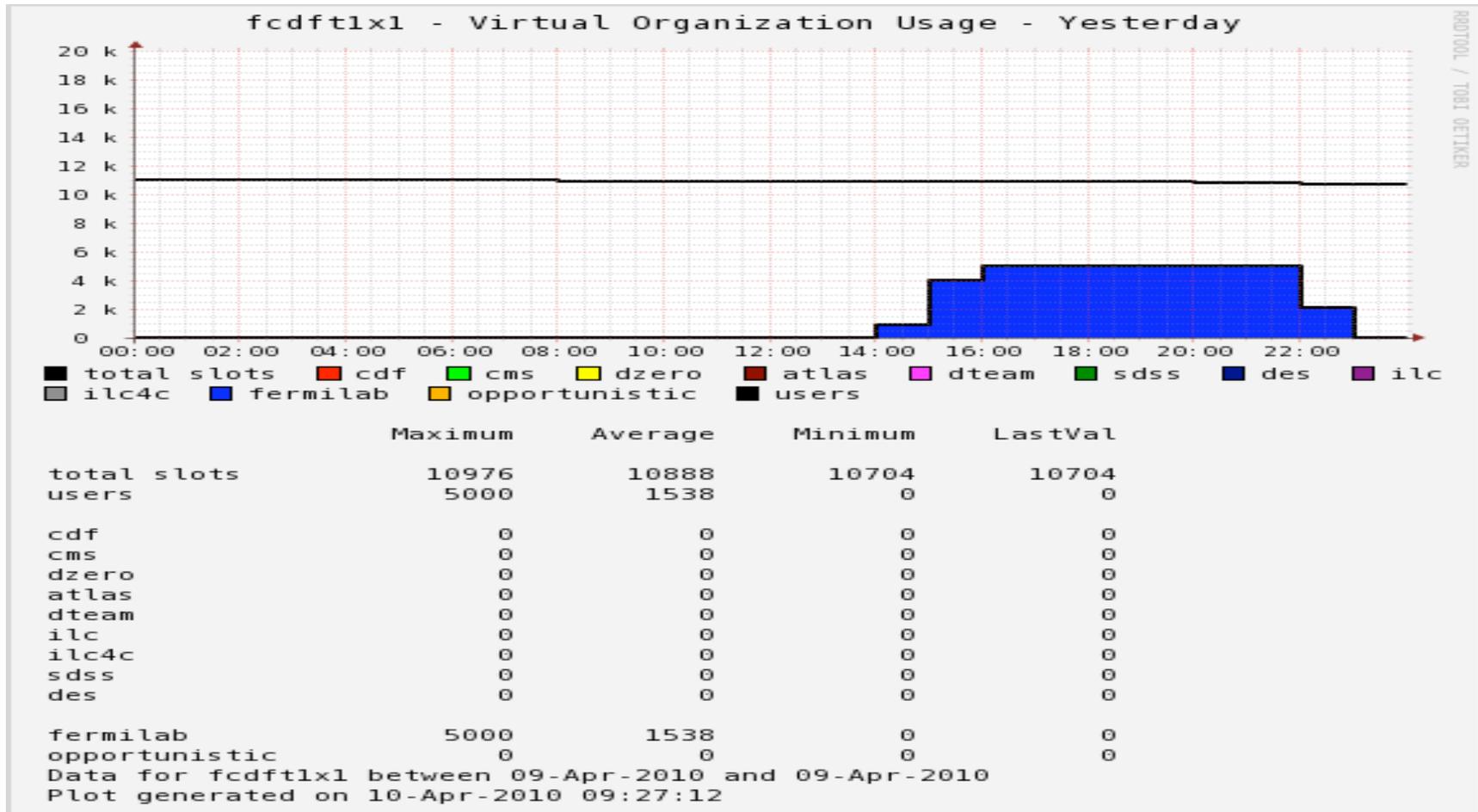
- Current SAZ (V2_0_1b) – fgt5x3:
 - Base=50x1x1000, Tsunami=1000x1x50
 - Immediately fail.
- New SAZ (V2_7_0) – fgt6x3:
 - Base=50x1x1000, Tsunami=1000x1x50
 - Ran for ~few minutes, then fail (“too many open files”).
- Fixed New SAZ (V2_7_0) – fgt6x3:
 - Base=50x1x1000, Tsunami=1000x1x50
 - Ran without incident
 - Average of 13.68 Authorizations/second.
 - Total elapsed time was 11,205 seconds (3.11 hours).
 - Base=50x1x1000, Tsunami=5000x1x50
 - Ran without incident
 - Average >15 Authorizations/second, Peak >22 Authorizations/second.
 - Total elapsed time was ~8 hours to process base+tsunami load.

SAZ Tsunami Profile (5000x1x50)

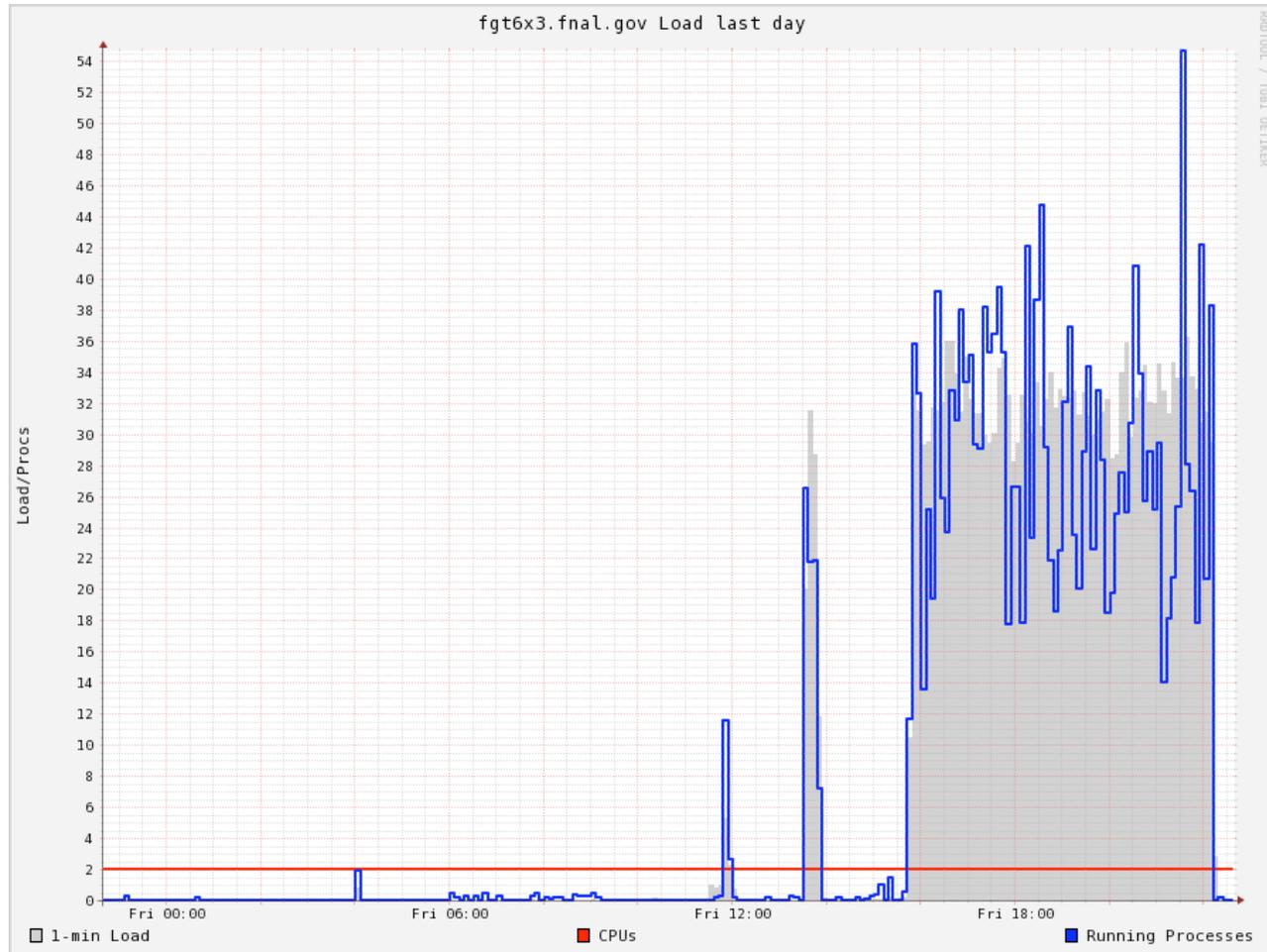




Another Tsunami (5000x1x50)



Load on fgdevsaz.fnal.gov



“Real World” Scale Tsunamis

- The previous tests using $5000 \times 1 \times 50 = 250,000$ authorization tsunamis are well beyond the actual real world experience.
- Most authorization tsunamis have been caused by a user issuing a “condor_rm” on a set of glide-in jobs.
- So comparison tests of fgdevsaz and fgt5x3 were run on a “real world” scale authorization tsunami – $5000 \times 1 \times 5 = 25,000$ authorizations.

fgt5x3 - tsunami 5000x1x5

- Black = number of condor jobs
- Red = number of saz network connections
- Trigger @ 11:45:12
- Failures start @ 11:45:19
- 25,000 Authorizations
- 14,183 Success
- 10,817 Failures
- Complete @ 11:58:28
- Elapsed time 13m 16s



fgtdevsaz - tsunami 5000x1x5

- Black = number of condor jobs
- Red = number of saz network connections
- Trigger @ 00:46:20
- 25,000 Authorizations
- 25,000 Success
- 0 Failures
- Complete @ 01:05:03
- Elapsed time = 18m 43s
- 22.26 Authorizations/sec



What's Next?

- Formal Change Management Request
 - Risk Level 4 (Minor Change).
- For build & test, we propose to deploy the Fixed New SAZ service (V2_7_0) on the pair of dedicated CDF Sleeper pool SAZ servers. This will allow us to benchmark the Fixed New SAZ service on a deployment that substantially matches the current production SAZ service (LVS, redundant SAZ servers).
- For release, we propose to upgrade one SAZ server at a time:
 - VM Operating System “upgrade” from 32 bit to 64 bit.
 - Install of Fixed New SAZ server (V2_7_0).
 - Verify functionality before going to the next.

Proposed Schedule of Changes

SAZ Servers	Server 1	Server 2
CDF Sleeper Pool	27-Apr-2010	28-Apr-2010
CMS Grid Cluster Worker Node	11-May-2010	12-May-2010
CDF Grid Cluster Worker Node	12-May-2010	13-May-2010
GP Grid Cluster Worker Node	18-May-2010	19-May-2010
D0 Grid Cluster Worker Node	19-May-2010	20-May-2010
Central SAZ Server for Gatekeepers	25-May-2010	26-May-2010

Note 1: All of the above dates are *tentative*, subject to approval by the Change Management Board and the corresponding stakeholder.

Note 2: FermiGrid-HA will maintain continuous service availability during these changes.